;login:
THE UNIX NEWSLETTER

Volume 5 Number 9 November 1980

CONTENTS

* UNIX is a trademark of Bell Laboratories

Guidelines for Submission of Newsletter Material

I would like to use the modern text preparation and communications facilities of UNIX to as great an extent as feasible in the preparation of the Newsletter. I have established an account on our PWB/UNIX system so that those who can provide us with machine manageable material can do so. The telephone number is (512) 474-5511. The login name is login and the password is usenix. (The system is also host utexas on the ARPANET.)

For those submitting paper copy of material, please produce your copy on a daisy wheel printer or similar high quality printing device. Line printer produced copy is typically not adequate for reproduction. Copy should be on 8 1/2" by 11" paper with a 1" margin on left, right, and bottom and 1 1/2" margin on top.

U. S. Mail submissions should be addressed to:

Login Newsletter
Computation Center
The University of Texas
Austin, Tx 78712

Attn: Wally Wedel

Announcements

USENIX Winter '81 Conference

Wednesday, January 21st, 1981
through
Friday, January 23, 1981

Software Tools User's Group Meeting

Tuesday, January 20th, 1981

The Jack Tar Hotel
Van Ness and Geary Streets
San Francisco, CA

Official notice is hereby given of the next national USENIX conference and Software Tools User's Group Meeting. The two conferences will be a total of 4 days longs and will consist of the following tentative agenda:

Tuesday, Jan 20	9-5 pm	Software Tools
Wednesday, Jan 21	9-11 am	USENIX registration
	10-12 noon, 1-5 pm	Technical presentations
	6-8 pm	Conference reception
Thursday, Jan 22	9-12 noon, 1-5 pm	Technical presentations
	7-10 pm	Vendor exposition & SIG groups
Friday, Jan 23	9-12 noon, 1-5 pm	Technical presentations

USENIX registration fees are:

	Regular	Student
Pre-registration	$30	$15
On-site registration	$80	$30

Completed pre-registration form and payment MUST be postmarked by 14 January 1981. Anything postmarked later will be charged full fees. Needless to say, pre-registration is strongly encouraged.

A vendor exposition is planned at this conference. BBN will have their C-70 "C" machine and Onyx will have one of their Unix systems running. Other interested vendors are urged to contact Tom Ferrin for space arrangements.

Technical presentations:

Abstracts are now being accepted from individuals wishing to make a technical presentation at the conference. Potential speakers must submit an abstract (100-200 words), preferably via electronic means, to Mike O'Dell at Lawrence Berkeley Labs (see below for address). An agenda will be

distributed at the meeting and abstracts must be screened in advance in order for individuals to make a presentation.

Possible topic areas include:

VAX	Status reports, what about the 750? Conversion efforts, performance issues, networking?
V7	Status reports, scaling for small systems, performance problems, conversion issues
Networks	NON-ARPAnet networks, USENET and UUCP efforts, CSNET? Arpanet issues for Vax and V7 systems, networking with alien hosts (IBM rje, CDC Hyperchannel, etc.)
Electronic Offices	Mail systems, editors, text processing, document compilers, forms management systems, "office information systems", high-performance output devices, typesetter simulators, USER EXPERIENCES!! !
Graphics	Systems, implementations (SIGGRAPH Core), integrated graphics and text for document generations, Unix architecture issues which effect graphics systems design.
Languages	What, who, where, why. Wither ADA? New C compilers, new anything else compilers, Algol68 on the Vax, Lisp (Franz and his friends), Vaxima, Snobol systems, APL, status reports, efforts envisioned.
Database Systems	Who, what, where? Distributed and otherwise, prototype, demonstration, and production.
Standards	Why, when, who? The effect of an ARPA-supported version of Unix on the Vax, what should be considered for standards? Why should we be interested? Position papers, pontifications.
Hardware	Neat new hardware for C, small Unix systems, large Unix systems, impressive peripherals, unimpressive peripherals, USER EXPERIENCES, configuration issues, the third-party maintenance alternative, "standard" device drivers?
Applications	Unix as a systems base, performance, reliability, name recognition, building "real-world" application in "NON-Cobol", why is Unix good for system builders, why is it bad? Who should consider it and who should keep their distance?
Commercial Products	Organizations with commercial products based on Unix are encouraged to make presentations based on the following guidelines: Since this is a technical meeting, we require presentations in the general sessions to be

of technical nature. They should be presented by someone with a deep technical understanding of the product and free to discuss any reasonable technical questions he/she raises. An ideal presenter would be a designer or implementor. There will be a specific session for "sales" presentations where people interested in those aspects can interact. We wish to encourage exposure to commercial offerings, but at the same time, desire to respect the basically technical nature of the event.

Future Directions — Where are we (the Unix world) going, who is going with us, and who is doing the driving? Impacts of Unix as a de facto standard operating system, vendor accommodation of Unix (i.e., the much rumored DEC C compiler), how does this interact with the "standards" issue? What comes after Unix?

The Unix Milieu — Anything not mentioned or implied somewhere above. What are you doing that is interesting or might save someone else from reinventing the wheel?

Abstracts must contain the following information:

1. Full Human Name

2. Installation name (and optional description)

3. Official US Mail address (probably the installation + routing info required to outwit local human mailers)

4. US Mail address which is good for quick response if electronic mail is not available.

5. Electronic mail address if at all possible (ARPAnet, or UUCP path relative to ucbvax)

6. Telephone number which will get to the person at the installation, and what hours of the day it is reasonable to attempt this. If this is difficult or impossible, a phone number to an arbitrary place, likely to reach the person. Again, list the hours when this number applies.

7. Audio-visual equipment requirements. (35mm slide projector? overhead? 16mm film? other?)

A large effort will be made to notify all persons selected to make presentations before the conference dates. A screening committee composed of Mike O'Dell, Bill Joy and Tom Ferrin will choose from all submitted abstracts; talks already presented at previous meeting(s) are discouraged unless they contain significant new material or are of exceptional interest.

Mailing addresses:

1. Conference registration information:

Tom Ferrin

ucbvax!ucsfcgl!tef (via uucp)
csvax.ucsfcgl!tef@berkeley (via arpanet)
School of Pharmacy (via uncle sam)
University of California
San Francisco, CA 94142

2. Abstracts of presentations:

Mike O'Dell

ucbvax!mo (via uucp)
mo@lbl-unix (via arpanet)
CSAM 50B/3243 (via uncle sam)
Lawrence Berkeley Laboratory
University of California
Berkeley, CA 94720

Note: Electronic mail is preferred; please, no phone calls.

Also, this is expected to be a popular conference. We expect around 500 attendees for this meeting (biggest to date) and facilities are limited. Plan to beat the holiday rush by making both hotel reservations and conference pre-registration EARLY!

Software Tools User's Group
(Announcement and Call for Papers)

1. When and Where?

January 20th, 1980
Jack Tar Hotel
Van Ness and Geary
San Francisco, CA

2. Registration Fees:

	Regular	Student
Pre-registration	$10	$ 5
On-site registration	$20	$10

Completed pre-registration form and payment MUST be postmarked by 14 January 1981. Anything postmarked later will be charged full fees. Needless to say, pre-registration is strongly encouraged.

3. What is Software Tools about? The software tools are a package of license-free, UNIX-like utilities and system calls, written in Ratfor and designed to be ported to almost any operating system. Originating from Brian Kernighan and P. J. Plauger's book Software Tools, the enhanced package now includes programs such as screen editors, mail

systems, enhanced Ratfor preprocessors, an nroff-like text formatter, and a command line interpreter similar to the Unix shell, as well as many of the other utilities available with Unix.

The tools present a virtual operating system interface consisting of a virtual machine (system calls or "primitives"), utility programs, and a command language, thus achieving inter-system uniformity over a variety of operating systems.

4. Agenda: (Meeting begins at 9:00 AM)

Session I The Virtual Operating System Approach -- Dennis Hall, Lawrence Berkeley Laboratory, Chair Description of the design and implementation of projects using the virtual operating system approach

Break

Session II New and Enhanced Utilities -- Dave Martin, Hughes Aircraft, Chair Descriptions of the design and implementation of new software tools.

Lunch

Session III Implementation Issues -- Joe Sventek, Lawrence Berkeley Laboratory, Chair Primitives specification, network extensions, and other implementation issues, including descriptions of implementations on various machines

Break

Session IV Special Interest Group Meeting Ratfor SIG, Network SIG, Primitives SIG, Text SIG, plus an informal group meetings concerned with implementation of primitives on various operating systems

Session V Future Directions

Abstracts of all presentations are required and should be forwarded to:

George Pajari
Clarendon Datex Ltd.
Fourth Floor
73 Water Street
Vancouver, BC V6B 1A1 Canada
(604) 688-1515
or
Debbie Scherrer
Computer Science and Applied Mathematics Department
Lawrence Berkeley Laboratory
Berkeley, CA 94720
(415) 486-5881

Letters

26 November 1980

Login Newsletter
Computation Center
The University of Texas
Austin, TX 78712
Attn: Wally Wedel

Dear Wally

Let me join in the resounding applause for the return of ;login:. It has been greatly missed.

At the risk of beating an overbeaten issue into the ground, I would like to make comments on the issue of recruiting at USENIX meetings.

I, too, as an employer of Unix* programmers and operators, find active recruiting at USENIX meetings unacceptable, but I also remember the days when I moved across the ocean and needed a job when I arrived here in New York. I used an old issue of the newsletter which contained the addresses of members as a source of prospective employers, and indeed, was hired by the second one I contacted. I suspect it is not unusual for experienced Unix people to find themselves without a job, and this is certainly true for students who have actually pulled enough credits to be turned out of their schools with a scroll under their arms. Hence, my suggestion is to forbid recruiting, but permit people who are looking for jobs to post a note to that effect. I would then be free to take people to meetings without the fear that they would come back only long enough to resign and pack, and I might be able to fill a position myself without having "stolen" someone away from another Unix installation.

I hope the Association will take this one-way approach under consideration the next time the subject comes up.

Thank you.

Sincerely

Mark Pearson, Head
Computer Services
Yourdon inc., 1133 Avenue of the Americas, New York, N.Y. 10036 212 730-2670

Biosciences Data Centre
The University of British Columbia
2204 Main Mall
Vancouver, B.C., Canada V6T 1W5

(604) 228-6527

November 20,1980.

Wally Wedel
University of Texas
Computation Center
Austin, Texas
U S A 78712

Dear Wally:

I was happy to see a issue of LOGIN finally appear, but I was not happy with the contents. It has now been many years since a "decent" issue of login appeared. By "decent" I mean, one with something other than USENIX introspection, meeting announcements, and games playing. In particular, I want a means of exchanging ideas and information about the following:

1 bugs or design flaws in UNIX

2 suggested improvments in UNIX

3 co-ordination of system changes, so that we don't end up with as many different implementations as there are installations.

4 ideas or suggestions for new nifty tools in UNIX (or even just a one liner such as: "We have implemented a MVDIR command that lets one change the name of a directory, including its heirarchial level while protecting against loops".

I am also very unhappy with the time it has taken to get out the USENIX distribution tapes. At the last meeting I attended (Boulder) it was promised that the tapes would be distributed as soon as possible. We send in the request forms within days of receiving them (it was in May or June I think) and haven't heard a peep since then. Every few months I have called New York and been assured that they were just about to get the newsletter (and/or tapes) distributed. Then nothing happens.

My total budget for software (e.g. What is left over after paying for maintanence, supplies, repairs etc.) is only about $500/year. The $160 that I have invested in USENIX over the last two years has brought me nothing except aggrevation, and is a significant proportion of my budget.

Since USENIX has not provided the services that it promised I am requesting a complete refund of the $160 if we do not receive the distribution tapes by the end of 1980.

I am enclosing a copies of letters that I have sent to the Canadian UNIX group as an example of the sort of information I would like to see in LOGIN.

Sincerely,

W. E. Webb.
Systems Analyst

WEW:fmt

cc
Canadian UNIX User's Group
USENIX, New York

Biosciences Data Centre
The University of British Columbia
2204 Main Mall
Vancouver, B.C., Canada V6T 1W5

(604) 228-6527

June 2,1980.

The Editor
Canadian UNIX Users Group
Human Computing Resources Corporation
10 St. Mary Street
Toronto, Ontario

Dear Group:

I thought that I would mention some of the things that I have been doing since I last had a letter published in LOGIN (obviously quite a while since LOGIN hasn't published anything of interest except meeting information for several years).

System changes: I have changed our system in the following ways that might be of interest to others:

In preparation for getting a large (300 MB) disk drive I changed the disk allocation free mechanism to maintain the disk block free count in the superblock so as to speed up "df". It turns out that most of the time spent in "df" is taken up in the SYNC calls to update the superblock. I made these calls optional and "df" now runs much faster.

I changed "trap.c" so that "trap" instructions may be intercepted. There is a new system call "systrap" that accepts an address in user D space. If this address is non-zero during a system call, it tests that word and if it in turn is non-zero, treats the TRAP instruction as a bad system call. This allows one to write a program that traces what system calls another program makes, as well as being useful for running DOS, RT-11 or RSX programs under UNIX.

I changed the swap space allocation routines so that if we run out of swap space (we often do so on 2 RK05's with 9 or 10 users), it will remove any of the unused STICKY programs and try again. This makes the penalty for STICKY programs less severe.

I changed the line-printer driver so that it picks up duplicate series of characters (usually blanks) and feeds PUTC/GETC with a character plus a (negative) count. This speeded up the driver by about 50 percent when in plot mode.

Programs Available:

Fortran 77 ... My Fortran 77 compiler is available to any UNIX installation for a $50 distribution charge.

Basic ... I have an ANSI Basic (written in C), available for a $25 distribution charge.

NTP ... A version of TP written in C (from the UNIX library tape), modified to use floppy disks, automatically create directories, and protect files on extracts.

PP ... A plot filter for Printronix 300 line printer. Would have to be modified to run with non-UBC plot routines.

LISP ... A LISP interpreter written in C.

FAKERT11 ... An program that runs RT-11 programs under UNIX.

RTUNIX ... A subroutine library that lets you run most UNIX programs under RT-11 (but only use it for your own programs, not Bell's).

AR ... An interface to AR that allows both new and old AR formats.

COST ... Two routines that let you figure out system utilization and charge for it (requires a few system changes in EXIT to work).

FS ... File save program for Mag-tape. Allows access to multiple versions of files. Requires modified MT driver.

CCC ... Interface to CC that only recompiles those files that have been changed since the last compile. (like MAKE, but simpler).

ASA ... Converts ASA (Fortran) carriage control into normal ASCII control characters.

UNARCV ... Converts V7 archeives back into original V6 format.

FILECP ... Reads a V6 file system under RT-11 or UNIX V7 or whatever.

I am planning on putting together a UBC distribution tape if there are enough requests for it. Cost will be about $100 and it will include Fortran77 and Basic and just about anything else of interest done at UBC.

Sincerely,

W. E. Webb.
Systems Analyst

WEW:fmt

Biosciences Data Centre
The University of British Columbia
2204 Main Mall
Vancouver, B.C., Canada V6T 1W5

(604) 228-6527

November 5,1980.

The Editor
Canadian UNIX Users Group
Human Computing Resources Corporation
10 St. Mary Street
Toronto, Ontario

Dear Group:

I thought that I would mention some of the things that I have been doing since I last wrote to you:

System changes: I have changed our system in the following ways that might be of interest to others:

1 I changed ICHECK so that I could tell it about bad blocks. This prevents UNIX from using those blocks, as after an "icheck -s" the bad blocks are taken out of the free list. Warnings are printed if a bad block appears in the filesystem other than the freelist.

2 I have changed DUMP/RESTOR to block the tapes. If no blocking is specified the programs work normally. In fact, a blocked dump tape may be read via an unmodified restor if one unblocks it via dd first (but you need lots of space or two tape drives).

3 I have modified "rm" so that it does the -r recursive directory delete itself rather than using GLOB, this means that one no longer gets the "no match" or "arg list too long" error messages.

4 "rm" also prints out the mode using the same format as "ls" so that it is easier to read. It also uses "/dev/tty" to read the response, so that it works properly from shell files.

5 I have written a program called LONG which can be used together with find to execute commands. For example to print out the status of all directories on the file-system one could use:
find / -type d -a -print ^ long ls -ld

6 I have modified find to use the -lS stdio library so that it no longer does one byte writes to files (ever wonder why find ... ^ cpio takes so long?)

7 we now have a working RM04 equivalent disk and a driver for it is available from us as well. (an RM04/RM05 is a 300 MB RM02/RM03). (ours is a CDC 9766 with a Western Peripherals controller).

8 I have modified RC (ratfor) to work with our Fortran 77.

9 I have modified the shell (Yale version) in several ways: (a) it now invokes /bin/bye upon logout from the top-level shell, which does any cleanup required, and prints a logged off message. It also checks for a .bye file and starts a shell on it for the user's cleanup. (b) it defines $H as home-directory, $B as bin-directory, and $U as user-name (c) upon a "cd" command (with no arguments) it checks to see if a .mail file with non-zero size exists, and if so, prints out "You have mail".

DISTRIBUTION TAPE
I have put together a distribution tape containing almost everything we have done here, including: Fortran 77, Basic, Lisp, NTP, and modifications to standard UNIX utilities such as rm, dump/restor, etc. To obtain it send me:

1 a check for $100

2 a copy of your UNIX agreement (or the appropriate letter from Bell where a sub-licence is concerned).

3 a 9 track tape of at least 1200 feet, stating a preference for either 800 BPI or 1600 BPI

I will send by return mail (or close to it) a copy of our distribution in TAR format (including TAR in case you don't have it). The distribution tape is about 11,000 blocks long (but its blocked so it will fit on a 1200 foot tape).

Sincerely,

W. E. Webb.
Systems Analyst

WEW:fmt

DECUS UNIX SIG Progress

Despite comments in the previous two mailings that each would probably be the last that you'd receive until the SIG began publishing a newsletter, here's yet another.

Mike Dash is our newsletter editor, and hopes to have the first issue in the mail to the DECUS office by early December. However, you may not receive it until sometime in January, since it takes two weeks or so for DECUS to make copies, and newsletters aren't sent first class (not to mention the Christmas mail overload that the post office has to deal with!). So, since minipapers for the spring symposium are due by January 9, it seemed that it might be wise to do our own mailing now to discuss plans for Miami.

The "Introduction to UNIX" and "UNIX Workshop" sessions at San Diego were well attended, and will be repeated next spring. There will probably be a short roadmap session the first morning, and a SIG business meeting sometime later in the week. What we need now are volunteers for other sessions. Anything that other UNIX users (or anyone else, for that matter!) might be interested in is appropriate, so long as it deals with UNIX applications on DEC hardware, or areas such as software tools implementations on Digital operating systems.

It's clear that there's a high level of user interest in the areas that our SIG deals with. The success of the SIG will depend to a large extent on the level of user involvement, so please feel encouraged to participate in SIG activities in whatever way(s) you feel would be useful. Symposium sessions are, of course, an excellent way to participate. Share your insights and experiences with others!

If you haven't received the Call for Participation, contact the DECUS office soon, so that you'll have a chance to submit your minipaper before January 9.

One final note: This mailing list has now been forwarded to the DECUS office. If you're a DECUS member, you are now a member of the SIG. If you're not a member, you will be receiving a membership form, so that you can join DECUS and the SIG. Unless we encounter some unforseen difficulty in getting out the first issue of the newsletter, this will definitely (for real, this time!) be the last mailing you'll receive un~til then. Please note that only SIG members will receive the newsletter.

Mark Bartelt

Caltech 346-48
Pasadena, California 91125
213/795-6811 ext 2663

DECUS SIG Newsletter editor:
Mike Dash
John Fluke Mfg. #25A
P.O. Box 43210
Mountain Lake Terrace, WA 98403
(206) 774-2390

Tiny C*

Tiny-C Two - The Compiler

Tiny-c two is ten times faster than tiny-c one. It has many extra features, including long (32 bit) integers, lots of new operators, and redirectable and direct access input/output. This version of tiny-c is viable for professional work, either systems programming or business applications.

It comes with a UNIX* style command interpreter called the "tiny-shell"*. With the tiny-shell, every compiled tiny-c program becomes a new shell command. Tiny-shell commands can have arguments, and dash (-) options, just as real UNIX shell commands do. The <and> input/output redirection operators are supported.

There are over fifty standard library functions, and this set is readily extended. The input/output functions are UNIX style, including fopen, fprintf, etc. Both ascii and raw (binary) input/output are supported.

And the entire package is portable. Bringing it up on a new processor or new operating system should take a few days or a few weeks at the most. And as usual with tiny-c products, all the source code is included.

Language Features

- All the features of tiny-c one
- Additional operators: not, complement, address of, postfix and prefix increment and decrement, left and right shift, and, or, exclusive
- UNIX style i/o; redirectable by the tiny-shell or by program, ascii and raw (binary), formatted print and scan, direct access (lseek)
- Program chaining for very large applications
- Dynamic storage allocation (calloc, cfree)
- Improved machine language interfaces

Physical Features

- 32K recommended. This is enough to compile the compiler.
- The compiler is written in tiny-c; all source code is included
- Emits a very compact, stack oriented intermediate code
- Interpreter for the intermediate code uses about 2K bytes
- Standard assembly language portion of the library uses about another 2K bytes. (The tiny-c coded portion of the library is loaded as needed).
- PORTABLE - readily transported to other processors or operating systems. The bootstrap procedure is well documented, and tests are

* UNIX is a trademark of Bell Laboratories, Inc.
Tiny-c and tiny-shell are trademarks of tiny c associates

provided.
- Speed: 500 to 1000 statements per second on typical 2 MHz to 4MHz 8 bit processors

Human Features

- Thorough documentation: over 200 pages. This includes a tutorial walk-through, a reference chapter, a reference with examples on the tiny-shell, lots of sample programs (and they are useful ones), internals describing how the compiler and linker interface to the tiny-shell, and all the details on how to install this system on any computer
- The tiny-shell support multiple commands per line, input/output redirection, and has thorough error control. Most commands have UNIX style dash (-) options.

For more information contact:

tiny-c Associates
Post Office Box 269
Holmdel, New Jersey 07733
(201) 671-2296

The Usenix Association

PURPOSE: The Usenix Association is an organization of Western Electric licensees and sub-licensees formed for the purpose of exchanging information and ideas about the UNIX operating system and the C Progamming Language.

MEMBERSHIP: Four classes of membership in Usenix are offered:

1. Institutional Membership. Institutional Members are the voting members of the Usenix Association. This class of membership is open only to licensees or sub-licensees of Western Electric Co.

2. Non-voting Institutional Membership. This class of membership is open to corporate affiliates of AT&T.

3. Individual Membership. Open to employees of class 1 and 2 members and others who are bound by the software agreements with Western Electric and its licensees.

4. Public Membership. Open to anyone with a bona fide interest in the purpose of the Usenix Association.

For further information write:

Usenix Association
Rockefeller University
Box 8
1230 York Avenue
New York, New York 10021
(212) 360-1182

Facts about UNIX and the Programming Language C

The UNIX operating system was developed by Ken Thompson and Dennis Ritchie of Bell Laboratories in Murray Hill, N.H., during the early 1970's. The C Programming Language was developed originally by Thompson and Ritchie as the implementation language for UNIX. UNIX is made available to the public by Western Electric Co. through its patent licensing office in Greensboro, North Carolina.

A fine overview of UNIX and C was published in the Bell System Technical Journal, Vol. 57, No.6; Part 2, in August 1978. The C Programming Language is described in the book The C Programming Language by Brian Kernighan and Dennis Ritchie published in 1978 by Prentice Hall.

Login Newsletter V5N9
Computation Center
The University of Texas at Austin
Austin, Texas 78712